

TIE-0250x concurrency mandatory projects

Project 1: Design, spring 2018

2018-03-02: more details on how the trains could/should run

2018-02-12: published

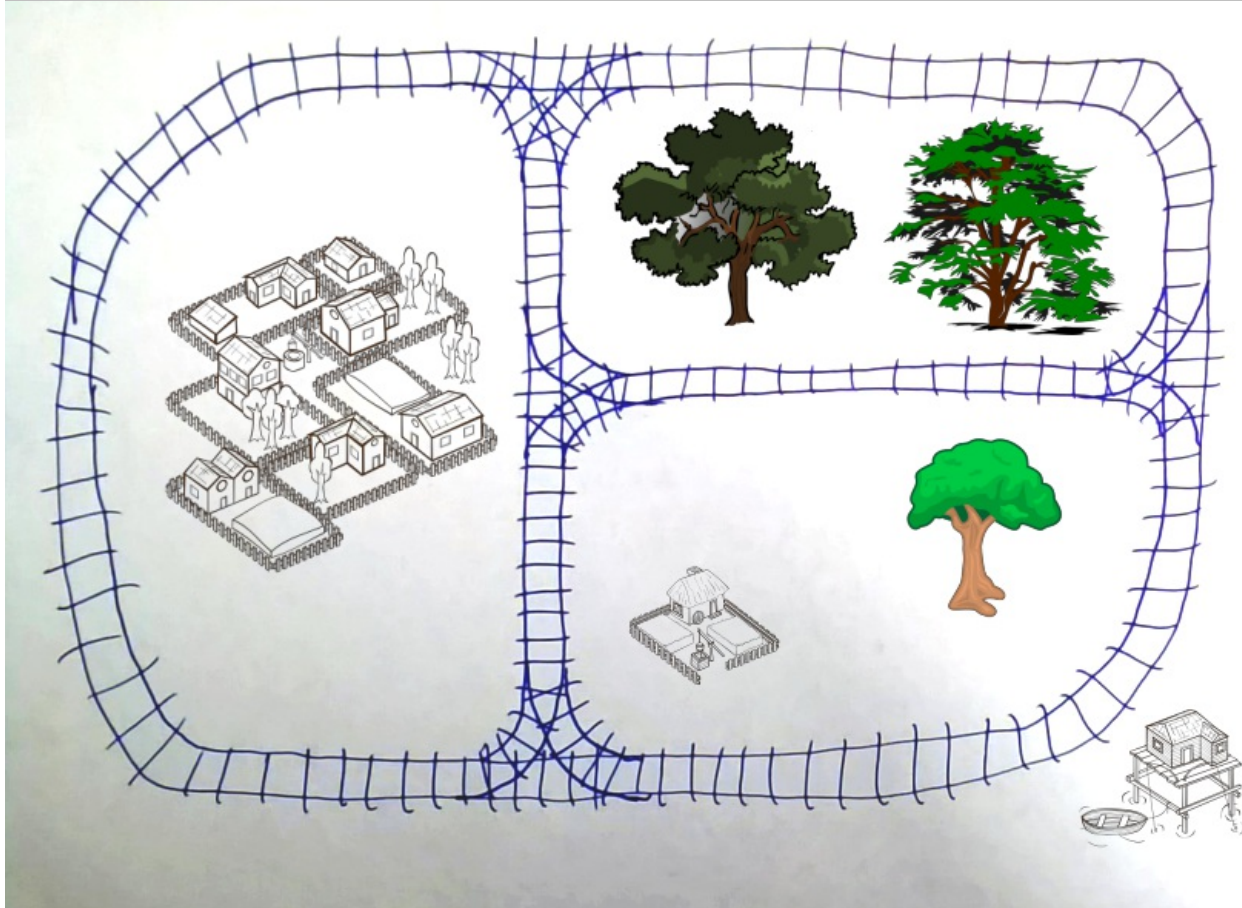
This document contains description for one project, remember also to read [the general information about the course projects](#).

1. Problem description

The main points in this project are:

- Analyse the problem space (what needs to be done)
- Design and document your solution so that the concurrent parts are easy and straightforward to implement using standard concurrent programming techniques.

2. Railroad simulation



The picture contains a railroad network and your task is to design simulation software, which runs 3-5 trains in this network.

Fixed point in design is that each train (locomotive) will be modelled by a thread of execution. The main objective of the project is to synchronise and manage the trains in such a way that, during the simulation, no collision or deadlock can occur (deadlock avoidance can be informal - you do not have to prove it works 100% of the time).

Boundary conditions

You are free to define the overall boundaries for your system, but these **MUST** be taken into account:

- Each track section is **ONE WAY** at a time: when a train has started going into one direction, there can not be any other trains going in opposite direction (note: this allows multiple trains in the **SAME** direction).
- When train has started on a section it **SHOULD** run to next section/junction (in other words: no U-turn in the middle of track section)
- Trains do **NOT** have to have a specific schedule (their overall movements can be random in the whole track-network).
- The track exchange/switching locations can be "magic" in the sense that any number of trains can be on them (waiting), all

trains can pass each other to any direction inside them.

Document

Write a design document, which outlines the structure of the simulation program. Concentrate on the concurrent programming issues (shared data, synchronisations). Description and figures are enough - you don't have to write any code in this project.

Concentrate first on identifying and documenting the critical parts of the system related to the concurrent simulation implementation. E.g. "the system SHOULD NEVER...", "system SHOULD do...".

THEN outline a solution to these parts. You DO NOT have to have code or even algorithm e.g. on how the trains run, description of the principles is enough (in this project).

Submission

Create ONE document in PDF format and save it to your group's GIT. General submission rules define how you submit your work from GIT to the Repolainen submission system.

3. Problems?

Send any questions about the project to the course email: rinn@tut.fi